What is claimed is:

1. A method of controlling a user application program executed in a client computer, comprising the steps of:

receiving a file readout request for a file from the user application program;

determining whether the file is stored in the client computer;

if it is determined that the file has been stored in the client computer, forwarding data of the file to the user application program; and

if it is determined that the file has not yet been stored in the client computer, receiving some of the data of the file from a predetermined server with the file stored therein and storing the some of the data in the client computer and forwarding the received data to the user application program, the predetermined server being connected to the client computer through a network.

2. The method as claimed in claim 1, wherein the step of receiving some of the data is performed by using data offsets of the file and the size of some of the data to be received.

3. The method as claimed in claim 1, further comprising the step of continuously executing the user application program in a state where only some of the data are received but all data of the file have not yet been received.

4. The method as claimed in claim 1, wherein the step of receiving the file readout request for the file from the user application program comprises the step of hooking the file readout request or mapping an original function for processing the file readout request to another function.

5. The method as claimed in claim 1, further comprising the step of receiving the data of the file from the predetermined server with the file stored therein and caching the received data in the client computer.

6. The method as claimed in claim 1, further comprising the steps of:

identifying a second client computer with the file stored therein; and

receiving the file from the identified second client computer and transferring the received file to the user application program.

7. The method as claimed in claim 1, further comprising the step of:

while the file readout request is not received from the user application program, receiving data expected to be required by the user application program from a second client computer with the data stored therein and storing the received data.

8. The method as claimed in claim 7, wherein determination on the data expected to be required by the user application program is made based on data that have been requested by the user application program upon previous execution of the user application program.

9. A method of controlling a user application program executed in a client computer, comprising the steps of:

receiving a file-writing request for a file from the user application program;

determining whether it is necessary to upload the file to a predetermined server that is connected to the client computer through a network;

if it is determined that it is necessary to upload the file to the server, uploading the file to the server; and

if it is determined that it is not necessary to upload the file to the server, writing the file in the client computer.

10. A method of controlling a user application program, comprising the steps of:

receiving a file readout request for a file from the user application program;

checking the user application program and data associated therewith, which are stored in a client computer, in response of the file readout request; and

if it is confirmed from the check that the user application program and the data have been altered, notifying a predetermined server of the alteration, the predetermined server being connected to the client computer through a network.

11. The method as claimed in claim 10, wherein the checking step is performed by using CRC check.

12. A method of controlling a user application program, comprising the steps of:

receiving a file readout request for a file from the user application program;

performing predetermined authentication while connecting with a predetermined server at a predetermined time interval; and

if the authentication fails, causing the user application program not to be executed by ignoring the file readout request.

13. The method as claimed in claim 12, wherein fees for the use of the user application program are charged to a user on the basis of pay-per-minute billing.

14. A method of controlling a user application program executed in a client computer, comprising the steps of:

storing at least one of a plurality of data blocks in a data file, which is accessed by the user application program, as a priority file in a predetermined server;

storing data offsets for the data blocks, sizes of the data blocks and priority file identifiers associated with the data blocks in an index storage means;

receiving a data readout request for an arbitrary data block in the data file from the user application program;

identifying a priority file corresponding to the arbitrary data block by referring to the index storage means; and

receiving the identified priority file from the predetermined server and transferring the priority file to the user application program.

15. The method as claimed in claim 14, further comprising the steps of:

identifying a second client computer with the priority file stored therein; and receiving the priority file from the identified second client computer.

16. The method as claimed in claim 14, further comprising the step of:

while the file readout request is not received from the user application program, receiving a priority file expected to be required by the user application program from a second client computer with the priority file stored therein and storing the received priority file.

17. The method as claimed in claim 16, wherein determination on the priority file expected to be required by the user application program is made based on priority files that have been requested by the user application program upon previous execution of the user application program.

18. A computer-readable recording medium on which a program for performing a method according to any one of claims 1 to 17 is recorded.